АЛГОРИТМ РАЦИОНАЛЬНОГО РАЗМЕЩЕНИЯ ФАЙЛОВ РАСПРЕДЕЛЕННОЙ БАЗЫ ДАННЫХ ПО УЗЛАМ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ С КОЛЬЦЕВОЙ ТОПОЛОГИЕЙ

Ладыженский Ю.В., Бельков Д.В.

Кафедра ПМиИ ДонГТУ

ladyzhen@pmi.donetsk.ua

Abstract

Ladyzhensky Y.V., Belkov D.V. The rational network database files location algorithm for the network with ring topology. *The effect network need rational network database files location. Heuristic algorithm of solution this task propose. Experiment results and analyzes for the network with ring topology adduce. There are mathematics model and database location files algorithm with files redistribution procedure. Heuristic algorithm of solution this task without files redistribution procedure propose .*

Введение

Рациональное размещение информации существенно влияет на эффективность функционирования вычислительной сети. Поэтому важно наилучшее размещение файлов распределенной базы данных(РБД) по узлам сети. В связи с большой размерностью задачи практически невозможно получение точного решения и необходима разработка приближенных эвристических алгоритмов. Этой проблеме посвящен ряд работ, отличающихся как постановкой задач, так и методами их решения [1,2,3,4].

В [4] рассмотрен эффективный подход к построению математических моделей оптимального распределения файлов РБД для сетей со звездообразной, кольцевой и произвольной топологиями. Особенностью, предложенного там общего алгоритма реализации моделей (далее алгоритм 1), является процедура перераспределения файлов. Необходимо, чтобы суммарного объема памяти узлов хватило для размещения файлов при их перераспределении. Иначе, если сеть имеет n узлов, то в случае, когда (n-1) узлов заполнено, алгоритм завершается при переполненном последнем узле.

В данной работе предлагается алгоритм без перераспределения файлов (далее алгоритм 2) для реализации моделей, рассмотренных в [4].

В качестве критерия оптимальности распределения файлов в алгоритме 2 принимается, как и в алгоритме 1, сумма произведений объемов данных, составляющих запросы и ответы, порожденные работой системы в единицу времени, на расстояния, на которые эти данные передаются.

1. Постановка задачи

Рассмотрим вычислительную сеть с кольцевой топологией : все узлы сети связаны между собой единым дуплексным каналом передачи данных. Файлы содержатся в локальных базах узлов таким образом, что имеется только одна копия каждого файла. Запрос, инициированный в любом узле, предполагает доступ к отдельному файлу и все запросы из фиксированного узла к одному и тому же файлу имеют одинаковую длину и требуют для ответа одинакового объема данных .

Пусть m - количество файлов РБД, n - количество узлов сети; Fi-файл i; Kj - узел j; X - матрица размещения файлов: Xij=1, если файл Fi находится в узле Kj, иначе Xij=0.

Целевая функция :

$$L=\sum_{i=1}^{m}\sum_{j=1}^{n} Q_{is} X_{is} \to \min,$$

где $Q_{is}=\sum_{j=1}^{n} \lambda_{ij}(\alpha_{ij}+\beta_{ij})d_{js}$ (1);

λ_{ij} - интенсивность запроса к файлу Fi, инициированного в узле Kj; α_{ij} - объем запроса к файлу Fi, инициированного в узле Kj; β_{ij}- объем запрашиваемых данных при выполнении запроса к файлу Fi, поступившего в узел Kj; d_{js} - расстояние между узлами Kj и Ks : $d_{j-1,s}=d_{j,s+1}=1$; $d_{js}=\min(|j-s|,n-|j-s|)$.

Ограничения :

$$\sum_{j=1}^{n} X_{ij}=1,\ i=1...m; \quad (2);$$

$$\sum_{j=1}^{m} L_i X_{ij}\le b_j,\ j=1...n \quad (3);$$

где Li - объем файла Fi; bj - объем узла Kj.

Необходимо найти матрицу размещения файлов X, обеспечивающую минимум целевой функции (1) при ограничениях (2),(3).

2. Описание алгоритма 2

1. Формируем матрицу Q по формуле (1);
2. Задаем нулевые значения матрице размещения файлов X и матрице запретов размещения E;
3. Задаем нулевые значения вектору заполнения узлов T;
4. Изменяем матрицу Q : если $L_i > b_j$, то $Q_{ij} := R$ с произвольным $R>\max Q_{ij}$;
5. Полагаем min =Qi1 и p=1;
6. Среди не запрещенных узлов ($E_{ij}\neq 1$) для каждого файла Fi находим $Q_{ip}=\min(Q_{ij}),j=1...n$;
7. Если файл Fi помещается в узел Kp, т.е. $T_p+L_i\le b_p$, то размещаем его в узле Kp : $X_{ip}:=1$; $T_p:=T_p+L_i$.

   Если файл Fi не помещается в узел Kp, то :

   а) запрещаем этот узел: $E_{ip}:=1$;

   б) среди всех узлов находим первый не запрещенный узел Ks, в который помещается файл Fi;

   в) полагаем p=s и min=Qip;

   г) переходим к пункту 6.
8. Используя полученную матрицу X, вычисляем функцию цели по формуле (1).

На этапе 7б может оказаться, что все узлы запрещены для размещения файла Fi: Xij=0 для всех j=1-:-n. В этом случае возможно увеличить объем узла с минимальным Qij на Li и повторить вычисления с шага 1.

3. Результаты экспериментов

Рассмотрим решение следующих задач рационального размещения файлов РБД по узлам сети при общих условиях :

Матрица Q :

21.40	6.60	18.00
36.00	15.10	30.10
19.50	27.30	10.35
31.00	44.00	23.00
10.65	29.00	36.65
22.20	5.70	19.50
37.80	14.30	33.50
19.50	28.50	12.00

количество файлов: 8; *количество узлов:* 3; *объем файлов:* $l_1=8$; $l_2=7$; $l_3=6$; $l_4=5$; $l_5=4$; $l_6=3$; $l_7=2$; $l_8=1$.

Объем узлов сети указан в таблице 1.

Таблица 1.

Номер задачи	Объем узлов сети
1	12; 6;18
2	18;18;18

Для экспериментов использовалась ПЭВМ Pentium - 166, программы составлены на языке Turbo Pascal 7.1, операционная система MSDOS 7.1. Один шаг алгоритма означает одну операцию сравнения; ПП - алгоритм полного перебора. Результаты решения задач приведены в таблицах 2,3.

Задача 1 демонстрирует ограничения, характерные для алгоритма 1. Алгоритм 1 не может решить данную задачу так как объем узлов является недостаточным для перераспределения файлов и при «оптимальном» распределении узел 3 остался переполненным.

Алгоритм 2 решает задачу с ошибкой 25.63 % за 81 шаг, затратив 5.42 секунд. Значение целевой функции - 180.45, оптимальное значение - 144.00 Алгоритм полного перебора требует 29678 шагов и 230.75 секунд работы.

Таблица 2.

Номер алгоритма	Размещение файлов			Функц. Цели	Число шагов	Время реш.,сек	Ошибка %
	Узел 1	Узел 2	Узел 3				
1	-	-	-	-	-	-	-
2	F3;F4;F8	F5;F7	F1;F2;F6	180.90	81	5.42	25.63
ПП	F1;F5	F6;F7;F8	F2;F3;F4	144.00	29678	230.75	0

При решении задачи 2 алгоритмом 1 ответ получен за 46 шагов, время работы - 0.35 секунд. Ошибка по сравнению с полным перебором - 8.7 %, значения целевой

функции - 121.20 и 111.50. Полный перебор требует 29678 шагов и выполняется 230.75 секунд.

Алгоритм 2 решил задачу за 81 шаг, затратив 5.42 секунд. Ошибка этого алгоритма - 4.80 %, значение целевой функции - 116.90. Задача 2 показываетспособность предлагаемого алгоритма получать результат с точностью выше, чем у алгоритма 1 и существенно быстрее, чем по алгоритму полного перебора.

Таблица 3.

Номер алгоритма	Размещение файлов			Функц. Цели	Число шагов	Время реш.,сек	Ошибка %
	Узел 1	Узел 2	Узел 3				
1	F5;F7	F1;F2;F6	F3;F4;F8	121.20	46	0.35	8.70
2	F5	F1;F2;F6	F3;F4;F7 F8	116.90	81	5.42	4.80
ПП	F5	F1;F2;F7	F3;F4;F6 F8	111.50	29678	230.75	0

Заключение

На этапе распределения файлов по узлам сети время выполнения у алгоритма 2 больше, чем у алгоритма 1, за счет повышения точности решения. Важно отметить, что чем выше точность первоначального решения , тем реже придется выполнять перераспределение файлов в дальнейшем. Поэтому более высокая точность, обеспеченная алгоритмом 2, увеличивает время эффективной работы сети на длительном этапе ее функционирования .

Перспективной является разработка методов уточнения начального решения, полученного алгоритмом 2, и достижение оптимального распределения файлов РБД с их помощью. Алгоритм 2 можно также использовать для сетей с произвольной топологией.

Литература

1. Аббасов А. М. Оптимизация размещения информационных баз с копиями в сети ЭВМ.// Автоматика и вычислительная техника, 1988, N 4.- С. 71-75.
2. Зиновьев Э. В., Клименко С. А. Декомпозиционный метод оптимального размещения информационных ресурсов в сетях ЭВМ с зональной структурой. //Автоматика и вычислительная техника, 1987, N 6.-С. 54-60.
3. Ярмош Н. А., Журавский Е.М. Алгоритм оптимального размещения файлов данных в распределенной вычислительной системе.// Весцi АН БССР, сер. фiз.тэхн. наук.- 1984, N 1.-С. 78-83.
4. Цегелик Г. Г. Системы распределенных баз данных. - Львов: 1990. - 167 с.